MPICH-G2: A Grid-Enabled MPI

Nicholas Karonis

Northern Illinois University

Brian Toonen & Ian Foster

Argonne National Laboratory

Research funded in part by DOE, DARPA, NASA, NSF

Computational Grids

- Loosely integrated computing systems
  - Heterogeneity in architecture & mechanism
  - Lack of central policy or control
  - Dynamic, unpredictable behaviors
- Occur at multiple levels in infrastructure
  - Wide area computing systems (“The Grid”)
  - Local and campus area networks
  - Tomorrow’s parallel computers
- Multiple application motivations
  - Distributed by design or by implementation

the globus project
www.globus.org

Grid-Enabled Programming Tools

- Tools that simplify programming for grids
  - Implement high-level programming models
  - Simplify mapping from application level to grid level, and vice versa
  - Enable robust, high performance
- Multiple possible approaches
  - Advanced compilation techniques
  - Develop entirely new programming models
  - Retarget existing programming models 

MPICH-G2: A Grid-Enabled MPI

- A complete implementation of the Message Passing Interface (MPI) for heterogeneous, wide area environments
  - Based on the Argonne MPICH implementation of MPI (Gropp and Lusk)
- Globus services for authentication, resource allocation, executable staging, output, etc.
- Programs run in wide area without change
- Major rewrite relative to MPICH-G1, with greatly improved performance

Grid-based Computation: Challenges

- Locate “suitable” computers
- Authenticate with appropriate sites
- Allocate resources on those computers
- Initiate computation on those computers
- Configure those computations
- Select “appropriate” communication methods
- Compute with “suitable” algorithms
- Access data files, return output
- Respond “appropriately” to resource changes

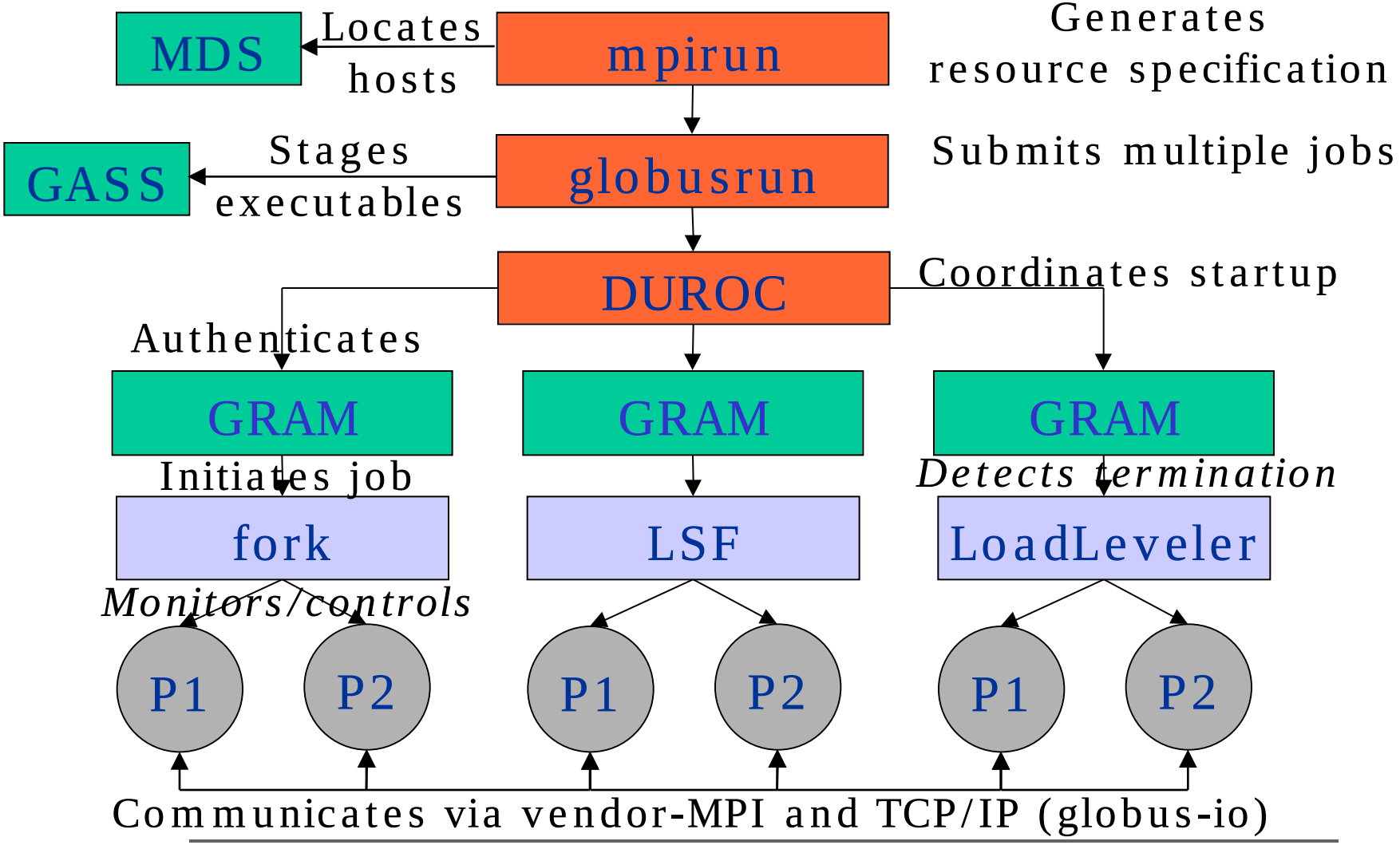
The Globus Toolkit

- A suite of "common grid services": security, allocation, communication, data access, information, fault detection, etc.
- Information-rich environment to enable automatic configuration and adaptation
- Translucent interfaces to allow tool-level and application-level optimizations
- A foundation on which can be built various higher-level libraries, tools, & applications

MPICH-G2 and Globus

% globus-proxy-init

% mpirun -np 256 myprog



the globus project
www.globus.org

Compound Resource Specification (To Become More User-Friendly!)

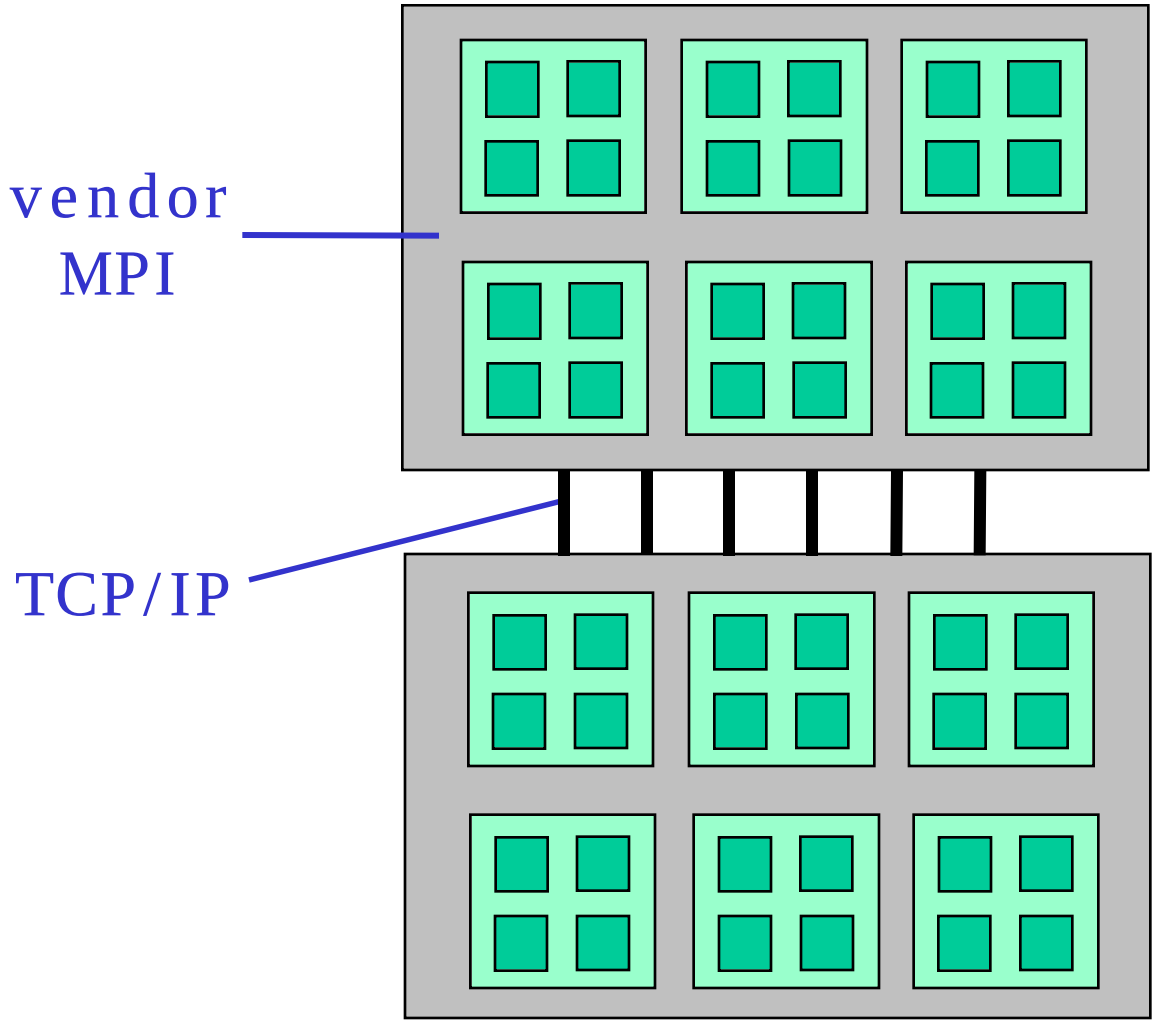
```
+( &(resourceManagerContact="flash.isi.edu")
    (count=1)
    (label="subjob 0")
    (environment=(FOO foo)(BAR bar)
                 (GLOBUS_DUROC_SUBJOB_INDEX 0))
    (arguments=arg1 arg2)
    (executable=/home/user/my_app1)
    (stdout=/home/user/my_app.out)
    (stderr=/home/user/my_app.err))
 &(resourceManagerContact="modi4.ncsa.uiuc.edu")
    (count=2)
    (label="subjob 1")
    (jobtype=mpi)
    (environment=(DISPLAY "modi4:0.0")
                 (GLOBUS_DUROC_SUBJOB_INDEX 1))
    (arguments=arga argb)
    (executable=/home/user/my_app2)
    (stdout=/home/user/my_app2.out)
    (stderr=/home/user/my_app2.err))
```

Different counts

Different executables

the globus project
www.globus.org

Multimethod Communication



- Detect transport mechanisms
- Select fastest

Performance Issues

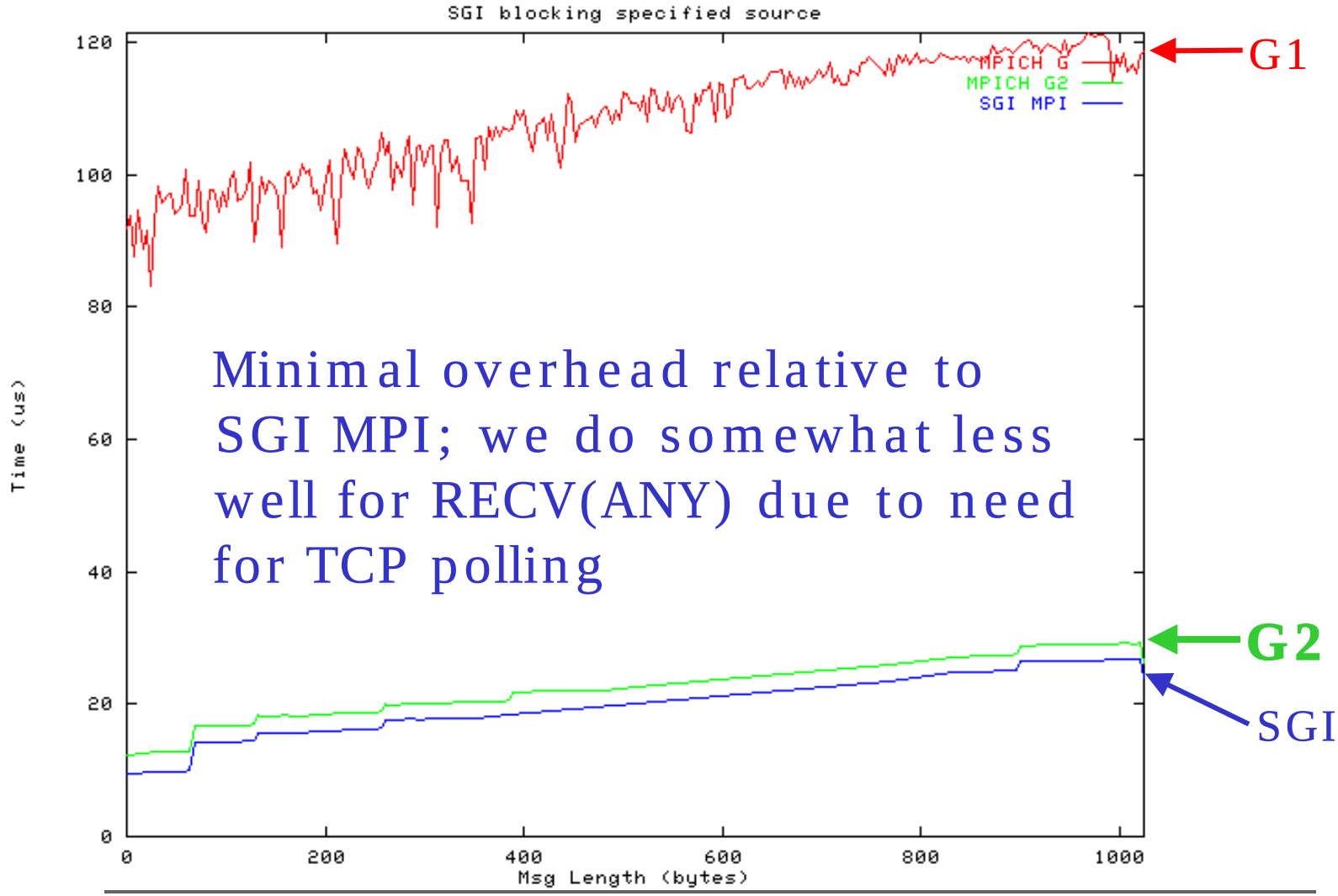
- Startup, monitoring, control
  - Scalability is fairly straightforward
- Communication
  - Managing multiple communication methods
  - Collective operations: topology-aware
  - Latency-tolerant mechanisms
  - Specialized wide area support: QoS, protocols

MPICH-G2 Experimental Studies

- Platform: SGI Origin 2000
  - Comparisons between SGI MPI, MPICH-G, and MPICH-G2
- Platform: 2 SUN workstations on LAN
  - Comparison between MPICH-P4, MPICH-G and MPICH-G2
- Focus to date is on microbenchmarks

the globus project
www.globus.org

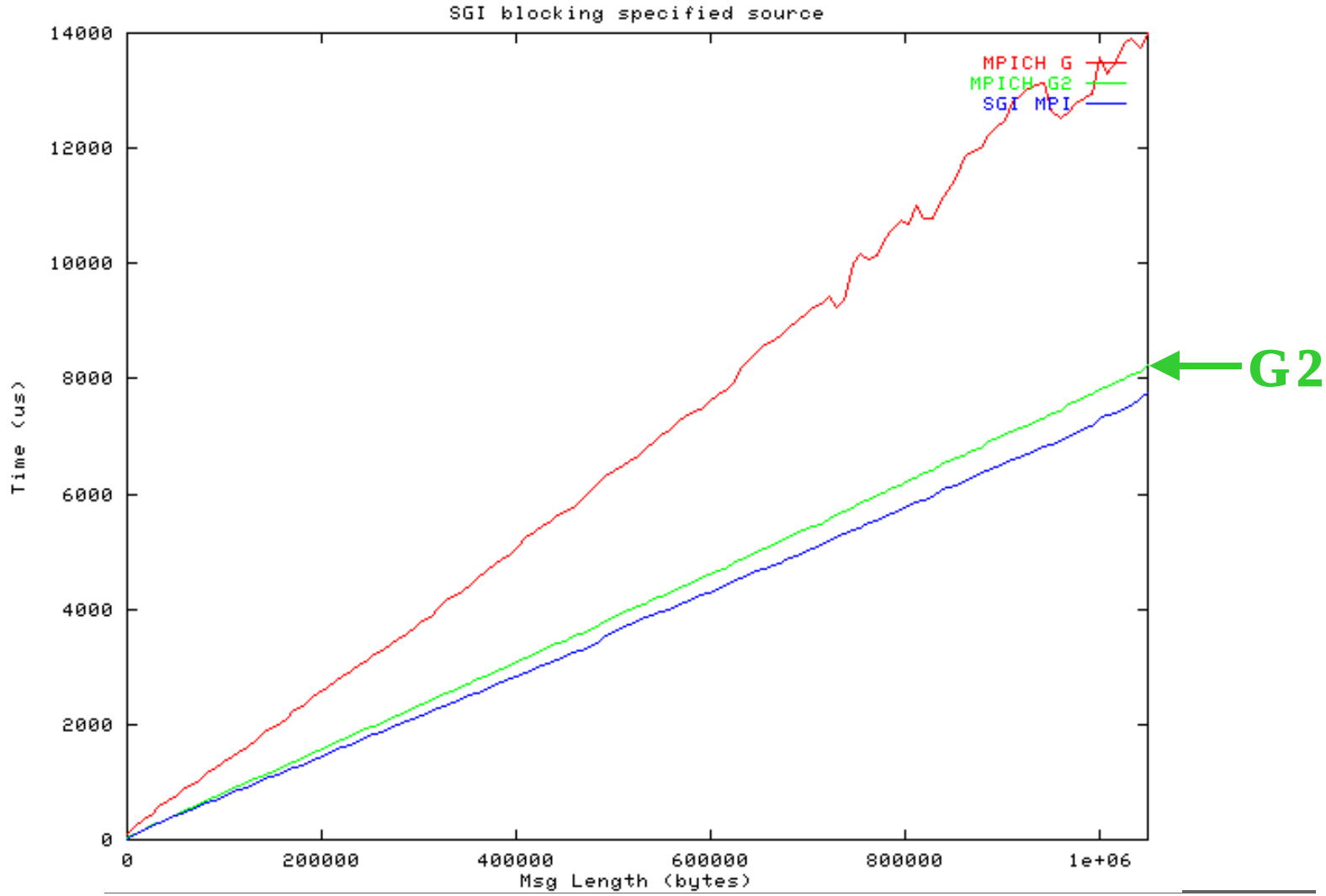
SGI Performance: Short Msgs



Minimal overhead relative to SGI MPI; we do somewhat less well for RECV(ANY) due to need for TCP polling

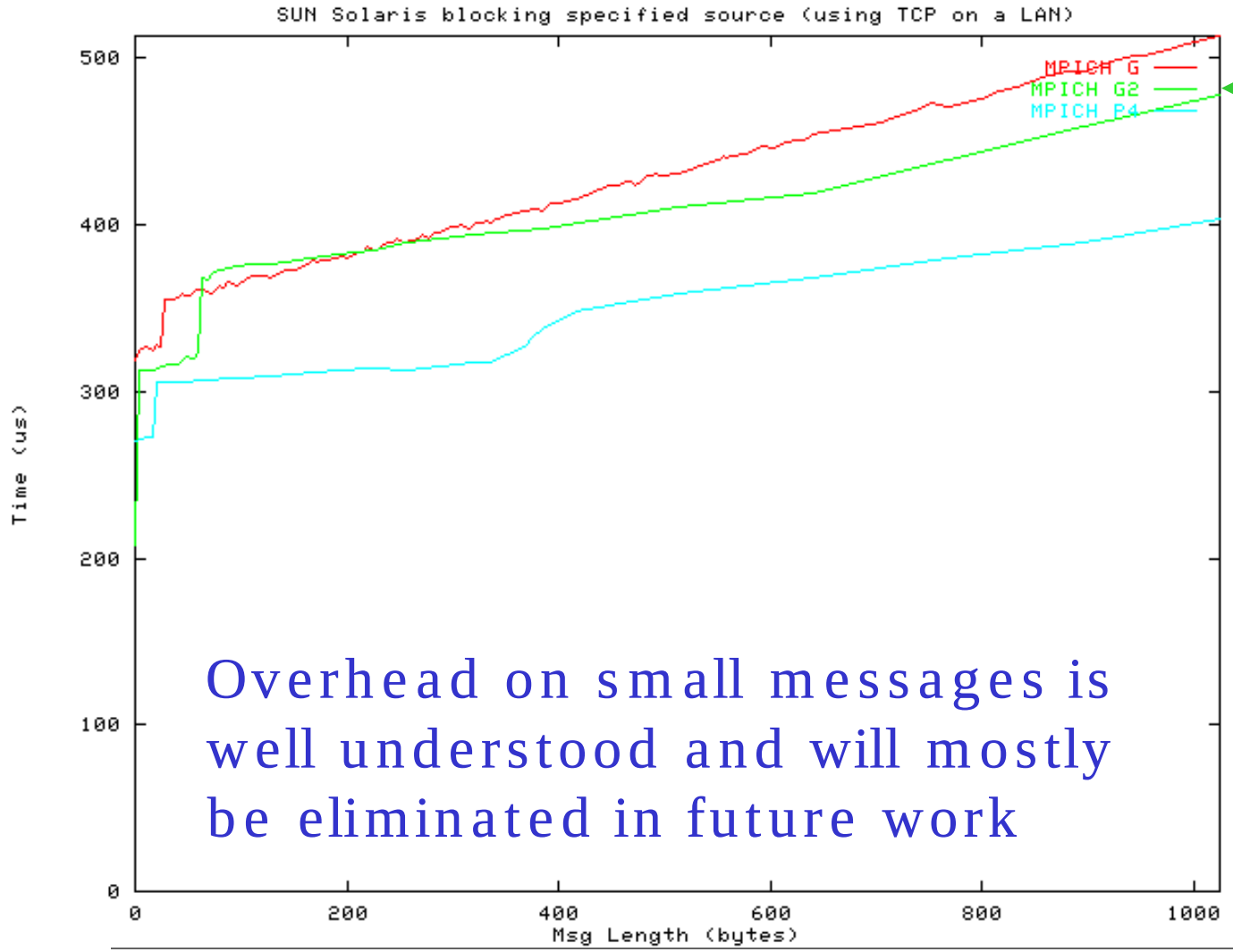
the globus project
www.globus.org

SGI Performance: Large Msgs



the globus project
www.globus.org

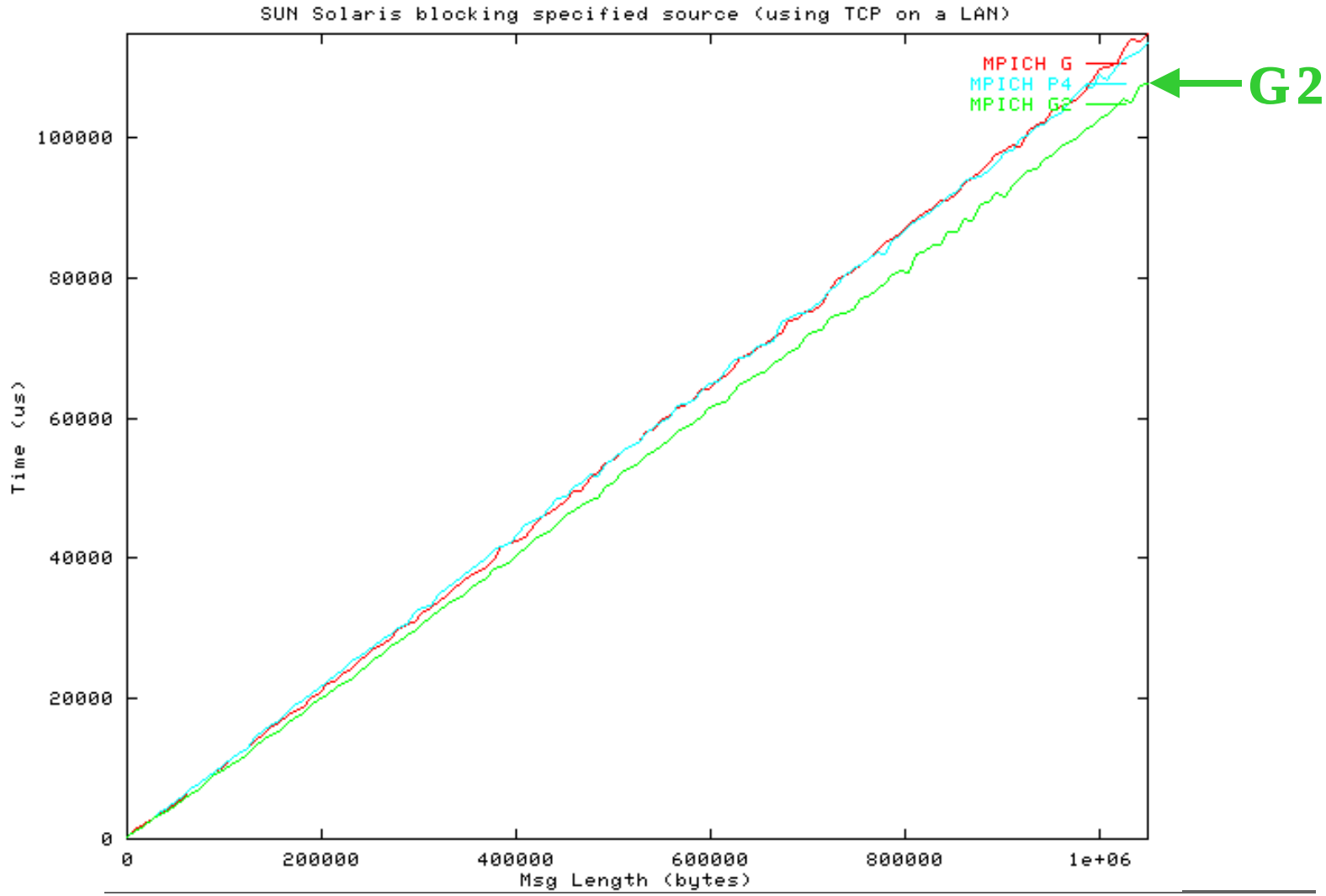
LAN Performance: Short Msgs



Overhead on small messages is well understood and will mostly be eliminated in future work

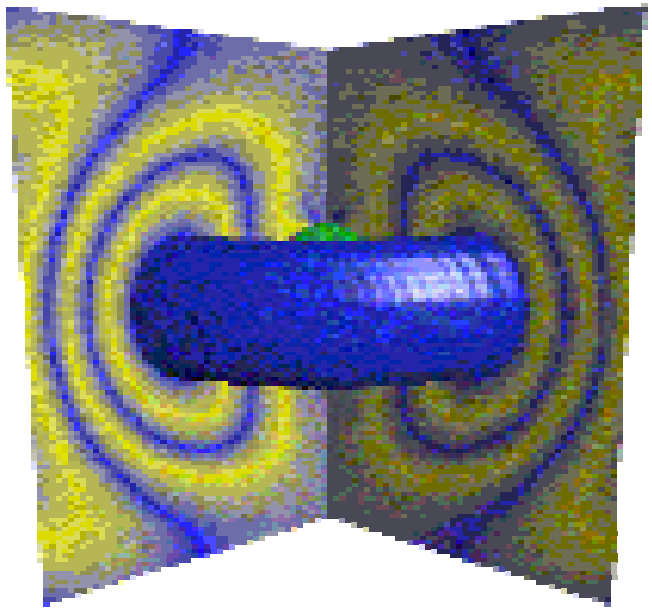
the globus project
www.globus.org

LAN Performance: Large Msgs



the globus project
www.globus.org

Example Applications



- Cactus (Max Planck Institute, Wash. U.)
  - Relativistic astrophysics
  - Experiments running across multiple supercomputers and multiple centers
- ScaLAPACK in testbed environment operated by NSF GrADS CS project

MPICH-GQ: QoS for MPI (Alain Roy, U.Chicago)

- MPICH-G2 integrated with GARA quality of service (QoS) architecture
- MPI programs manage QoS via communicator attributes
- MPICH-G2 impln uses GARA calls to invoke differentiated services mechanisms
- Study of QoS parameters needed for various MPI flows (typically large messages)
- Paper in Proceedings of SC'2000; see also http://www.mcs.anl.gov/qos

Related Work

- Cluster message-passing libraries
  - E.g., PVM, p4, LAM
- Wide area MPIs
  - E.g., PVMPI, PAC-X, STAMPI, MetaMPI; IMPI
- Our specific contributions
  - Faithful replication of single-machine model in heterogeneous, multi-domain environments
  - Integration with standard Grid services
  - High-performance multi-method comms
  - Grid-aware collective operations

MPICH-G2 Summary

- Functionally complete & correct MPI impln
- Performance excellent; additional optimization work proceeding
- A model for how to use Globus services to construct a “grid-enabled” version of an existing programming model and tool
- A powerful tool for Grid computing: but does not eliminate algorithmic challenges!
- For more information:

  http://www.globus.org/mpi